AW

UNITED STATES PATENT AND TRADEMARK OFFICE

UNITED STATES DEPARTMENT OF COMMERCE
United States Patent and Trademark Office
Address: COMMISSIONER FOR PATENTS
P.O. Box 1450
Alexandria, Virginia 22313-1450
www.uspto.gov

APPLICATION NO.	FILING DATE	FIRST NAMED INVENTOR	ATTORNEY DOCKET NO.	CONFIRMATION NO.
08/959,149	10/28/1997	RODNEY LIMPRECHT	3382-47280	4269

7590 07/08/2003

KLARQUIST SPARKMAN CAMPBELL
LEIGH & WHINSTON
ONE WORLD TRADE CENTER SUITE 1600
121 S W SALMON STREET
PORTLAND, OR 972042988

EXAMINER	
LAO, SUE X	
ART UNIT	PAPER NUMBER
2126	28

DATE MAILED: 07/08/2003

Please find below and/or attached an Office communication concerning this application or proceeding.

Office Action Summary	Application No. 08/959,149	Applicant(s) Limprecht, et al
	Examiner S. Lao	Art Unit 2126

-- The MAILING DATE of this communication appears on the cover sheet with the correspondence address --

Period for Reply

A SHORTENED STATUTORY PERIOD FOR REPLY IS SET TO EXPIRE 3 MONTH(S) FROM THE MAILING DATE OF THIS COMMUNICATION.

- Extensions of time may be available under the provisions of 37 CFR 1.136 (a). In no event, however, may a reply be timely filed after SIX (6) MONTHS from the mailing date of this communication.
- If the period for reply specified above is less than thirty (30) days, a reply within the statutory minimum of thirty (30) days will be considered timely.
- If NO period for reply is specified above, the maximum statutory period will apply and will expire SIX (6) MONTHS from the mailing date of this communication.
- Failure to reply within the set or extended period for reply will, by statute, cause the application to become ABANDONED (35 U.S.C. § 133).
- Any reply received by the Office later than three months after the mailing date of this communication, even if timely filed, may reduce any earned patent term adjustment. See 37 CFR 1.704(b).

Status

1) ☒ Responsive to communication(s) filed on *Apr 23, 2003* .

2a) ☐ This action is **FINAL**. 2b) ☒ This action is non-final.

3) ☐ Since this application is in condition for allowance except for formal matters, prosecution as to the merits is closed in accordance with the practice under *Ex parte Quayle*, 1935 C.D. 11; 453 O.G. 213.

Disposition of Claims

4) ☒ Claim(s) *1-5 and 7-28* is/are pending in the application.

4a) Of the above, claim(s) ______ is/are withdrawn from consideration.

5) ☒ Claim(s) *1-5, 7-12, 14-17, and 22-28* is/are allowed.

6) ☒ Claim(s) *13 and 18-21* is/are rejected.

7) ☐ Claim(s) ______ is/are objected to.

8) ☐ Claims ______ are subject to restriction and/or election requirement.

Application Papers

9) ☐ The specification is objected to by the Examiner.

10) ☐ The drawing(s) filed on ______ is/are a) ☐ accepted or b) ☐ objected to by the Examiner.

Applicant may not request that any objection to the drawing(s) be held in abeyance. See 37 CFR 1.85(a).

11) ☐ The proposed drawing correction filed on ______ is: a) ☐ approved b) ☐ disapproved by the Examiner.

If approved, corrected drawings are required in reply to this Office action.

12) ☐ The oath or declaration is objected to by the Examiner.

Priority under 35 U.S.C. §§ 119 and 120

13) ☐ Acknowledgement is made of a claim for foreign priority under 35 U.S.C. § 119(a)-(d) or (f).

a) ☐ All b) ☐ Some* c) ☐ None of:

1. ☐ Certified copies of the priority documents have been received.
2. ☐ Certified copies of the priority documents have been received in Application No. ______ .
3. ☐ Copies of the certified copies of the priority documents have been received in this National Stage application from the International Bureau (PCT Rule 17.2(a)).

*See the attached detailed Office action for a list of the certified copies not received.

14) ☐ Acknowledgement is made of a claim for domestic priority under 35 U.S.C. § 119(e).

a) ☐ The translation of the foreign language provisional application has been received.

15) ☐ Acknowledgement is made of a claim for domestic priority under 35 U.S.C. §§ 120 and/or 121.

Attachment(s)

1) ☒ Notice of References Cited (PTO-892)

2) ☐ Notice of Draftsperson's Patent Drawing Review (PTO-948)

3) ☐ Information Disclosure Statement(s) (PTO-1449) Paper No(s). ______

4) ☐ Interview Summary (PTO-413) Paper No(s). ______

5) ☐ Notice of Informal Patent Application (PTO-152)

6) ☐ Other:

DETAILED ACTION

1. Claims 1-5, 7-28 are pending. This action is in response to the amendment filed 4/23/2003. Applicant has amended claims 14 and 18.

2. The non-statutory double patenting rejection, whether of the obviousness-type or non-obviousness-type, is based on a judicially created doctrine grounded in public policy (a policy reflected in the statute) so as to prevent the unjustified or improper timewise extension of the "right to exclude" granted by a patent. *In re Thorington,* 418 F.2d 528, 163 USPQ 644 (CCPA 1969); *In re Vogel*, 422 F.2d 438, 164 USPQ 619 (CCPA 1970); *In re Van Ornum*, 686 F.2d 937, 214 USPQ 761 (CCPA 1982); *In re Longi*, 759 F.2d 887, 225 USPQ 645 (Fed. Cir. 1985); and *In re Goodman*, 29 USPQ2d 2010 (Fed. Cir. 1993).

A timely filed terminal disclaimer in compliance with 37 CFR 1.321(b) and © may be used to overcome an actual or provisional rejection based on a non-statutory double patenting ground provided the conflicting application or patent is shown to be commonly owned with this application. See 37 CFR 1.78(d).

Effective January 1, 1994, a registered attorney or agent of record may sign a terminal disclaimer. A terminal disclaimer signed by the assignee must fully comply with 37 CFR 3.73(b).

3. Claims 13 and 18-21 are rejected under the judicially created doctrine of obviousness - type double patenting as being unpatentable over claims 1-21 of U.S. Patent No. 5,890,161 to Helland in view of Steinman ("Incremental State Saving in SPEEDS Using C++") and Hutchison et al (U S 6,026,428). In particular, claims 1-21 of U.S. Patent No. 5,890,161 to Helland teaches server application (server application, claims 1, 7), application component (application component, claims 1, 7, 13), state (context object, claims 9, 13, 14, 16), function code (transaction, claims 1, 7, 13), operating service (run-time service, claims 9, 13), finishing transaction in response to an indication from the application component without action by client (indication from the application component

that its transactional work is complete, claims 7, 13, 14, 15), as recited in claims 13, 18 and 21 of the present application, and an indication from the application component that the component's work is complete (indication from the application component that its transactional work is complete, claims 7, 13; member function / abort function, claim 14), as recited in claims 13 and 18 of the present application.

Claims 1-21 of U.S. Patent No. 5,890,161 to Helland do not teach maintaining the state between method invocations, which is met by Steiman who teaches (SPEEDS system) maintaining an object state (v1) in main memory between method invocations (between events/messages/method calls, by delta exchange method), see sections 3, 4. In Steinman maintains object state without requiring an indication that work is complete. It would have been obvious to apply the teaching of Steiman to Helland because it improves resource efficiency by reducing the overhead of state savings (Steinman, page 695, left col., last para.).

Claims 1-21 of U.S. Patent No. 5,890,161 as modified do not explicitly teach destroying the state when a transaction is complete, although it would have been typical. Hutchison teaches destroying an object's state when its transactional work is finished (end association of context object with a thread/transaction when it reaches the end of its transaction lifecycle) (col. 3, line 55 - col. 4, line 30). It would have been obvious to apply the teaching of Hutchison to Helland because it allows multiple ways of handling contexts (col. 2, lines 1-25).

Regarding claims 19-20, holding a reference and releasing a reference are part of the conventional object creation and destruction. Further, Steiman teaches resetting the state (restore state by calling exchange again, section 4).

4. Claims 13, 18-21 are rejected under 35 U.S.C. 103(a) as being unpatentable over The Common Object Request Broker: Architecture and Specification CORBA (Revision 2.0) in view of Steinman ("Incremental State Saving in SPEEDS Using C++") and Coskun (U S Pat. 5,764,958)

As to claim 21, CORBA teaches (chapter 4, pages 12-16) server applications (servers, applications), executing (invoke) an application component (object) under control of an operating service (ORB), the application component having a state (context) and function code (method) for performing work responsive to a call (invoke method) from a client (client), destroying the state by the operating service (delete context object by CORBA::CTX_DELETE()). It is noted that a destroyed state in CORBA is not persistent.

CORBA does not teach (1) the step of maintaining, (2) destroying is in response to an indication from application component without action by the client.

As to (1), Steinman teaches (SPEEDS system) maintaining an object state (v1) in main memory between method invocations (between events/messages/method calls, by delta exchange method), see sections 3, 4. Given the teaching of Steinman, it would have been obvious to maintain an object state in main memory between method invocations. One of ordinary skill in the art would have been motivated to apply the teaching of Steinman to CORBA because it further improves the resource efficiency by reducing the overhead of state savings (Steinman, page 695, left col., last para.). It is noted that Steinman maintains object state without requiring an indication that work is complete.

As to (2), Coskun teaches destruction of a state of an application component (delete teacher object) is controlled by the server without waiting for consent from the client (when the call to server function returns). See col. 2, lines 14-19; col. 4, lines 31-36. The return from the call to the server function is the indication from the server object itself that its work is complete. The server/teacher object being no longer needed is also an indication that the server object's work is complete. Given the teaching of Coskun, it would have been obvious to destroy the state of the application component in response to an indication from application component but without action by the client. One of ordinary skill in the art would have been motivated to apply the teaching of Coskun to CORBA because it permits dynamic introduction of objects (Coskun, col. 1, lines 56-58) which is a primary purpose of CORBA.

As to claims 18 and 13, note discussion of claim 21 and note the equivalence of discarding / destroying, and before receiving / without action by the client. CORBA further teaches (chapter 4, pages 12-16) system service (ORB) for creating and destroying. CORBA teaches (chapter 2, page 9, section 2.1.11) instance creation service (object activation), client request (request), returning a reference (generate object reference). Typically in CORBA, a client calls a object / component function indirectly by calling a stub / object adapter to initiate work (invoke object) via the run-time/system service (ORB, including object adapter) and using the reference. CORBA further teaches encapsulating function code (object method) and a processing state (context) for the work in a component (context object), providing a reference (object reference) through an operating service (CORBA) for a client program to call the function code of the component to initiate processing (invoke method) (see discussion of claim 21 with respect to CORBA).

Regarding destroying/discarding processing state responsive to indication from the component that processing/work is complete and without action from the client, this is met by Coskun, as discussed for claim 21. The return from the call to the server function is the indication from the server object itself that its work is complete. The server/teacher object being no longer needed is also an indication that the server object's work is complete.

As to claims 19-20, holding a reference and releasing a reference are part of the conventional object creation and destruction. Further, CORBA as modified teaches (Steinman) resetting the state (restore state by calling exchange again, section 4). The factory mechanism of CORBA produces a component/object instance and its pointer. When an object is reused, its state is typically reset/reinitialized.

5. Claims 1-5, 7-12, 14-17 and 22-28 are allowed.

6. Applicant's arguments filed 4/23/2003 have been fully considered but they are not persuasive.

Applicant argued in substance that Coskun does not teach responsive to component's own indication (claim 21) and responsive to component's own indication that work is complete (claims 13 and 18). Remarks, pages 11-14, 15-17.

The examiner respectfully disagrees. Coskun teaches responsive to component's own indication because the server object (teacher object) is deleted after determining that the object is no longer needed and when the call to the server function returns. The return from the call to the server function is the indication from the server object itself that its work is complete. The server/teacher object being no longer needed is also an indication that the server object's work is complete. In fact, as disclosed, applicant's indications of work completion include a return from a call to the component. See, for example, application as filed, page 6, lines 12-14; page 34, lines 5-10. Therefore, Coskun meets responsive to components own indication and responsive to components own indication that work is complete both as claimed and as disclosed.

7. Any inquiry concerning this communication or earlier communications from the examiner should be directed to Sue Lao whose telephone number is (703) 305-9657. A voice mail service is also available at this number. The fax phone numbers for the organization where this application or proceeding is assigned are (703) 746-7238 for After Final communications, (703) 746-7239 for Official communications and (703) 746-7240 for Non-Official/Draft communications.

Any inquiry of a general nature or relating to the status of this application or proceeding should be directed to the Group receptionist whose telephone number is (703) 305-9600.

Sue Lao
June 26 2003

Sue Lao